Crank-Nicolson

The idea of the Crank-Nicolson method [1], is to discretise both the time and space domains and to use a combination of the forward and backwards Euler methods to compute each timestep.

Finite differences for time

The Schrödinger equation can be written in the form:

$$\frac{\partial \Psi(x,t)}{\partial t} = \frac{i\hbar}{2m}\frac{\partial^2 \Psi(x,t)}{\partial x^2} - \frac{i}{\hbar}V(x)\Psi(x,t) = F(x,t)$$

The quick and dirty way to derive the CN method is to simply integrate both of the sides of the equation:

$$\begin{aligned}
\int_{n\Delta t}^{(n+1)\Delta t} dt \frac{\partial \Psi(x,t)}{\partial t} &= \int_{n\Delta t}^{(n+1)\Delta t} dt F(x,t) \\
\Psi(x,(n+1)\Delta t) - \Psi(x,n\Delta t) &= \int_{n\Delta t}^{(n+1)\Delta t} dt F(x,t) \\
\Psi(x,(n+1)\Delta t) - \Psi(x,n\Delta t) &= \frac{\Delta t}{2}\left[F(x,(n+1)\Delta t) + F(x,n\Delta t)\right] + \mathcal{O}((\Delta t)^2) \\
\frac{\Psi(x,(n+1)\Delta t) - \Psi(x,n\Delta t)}{\Delta t} &\approx \frac{1}{2}\left[F(x,(n+1)\Delta t) + F(x,n\Delta t)\right].
\end{aligned}$$

The integral is approximated with the trapezoid rule (with error proportional to the square of the timestep).

A more rigorous way to derive the method gives insight to the errors that are intrinsic to the method. The position is evaluated at integer multiples of the stepsize $i\Delta x$, while the time is evaluated at the midpoint between integer steps $(n+\frac{1}{2})\Delta t$. Taylor series centered at the halfwaypoint $t = (n+\frac{1}{2})\Delta t$ are used to approximate $\partial_t \Psi$ and $F(x,t)$. To shorten the notation, expressions $t_n = n\Delta t$, $t_{n+\frac{1}{2}} = \left(n+\frac{1}{2}\right)\Delta t$ and $t_{n+1} = (n+1)\,\Delta t$ are used going forward.

Taylor series centered around point a is defined as:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n.$$

Taylor series (only with respect to time, fixed position) for $\Psi(t_{n+1})$ and $\Psi(t_n)$ centered around $t_{n+\frac{1}{2}}$:

$$\Psi(t_{n+1}) = \Psi\left(t_{n+\frac{1}{2}}\right) + \frac{1}{2}\Psi'\left(t_{n+\frac{1}{2}}\right)\Delta t + \frac{1}{8}\Psi''\left(t_{n+\frac{1}{2}}\right)(\Delta t)^2 + \dots$$
$$\Psi(t_n) = \Psi\left(t_{n+\frac{1}{2}}\right) - \frac{1}{2}\Psi'\left(t_{n+\frac{1}{2}}\right)\Delta t + \frac{1}{8}\Psi''\left(t_{n+\frac{1}{2}}\right)(\Delta t)^2 + \dots$$

Using these representations:

$$\Psi\left(t_{n+1}\right) - \Psi(t_n) = \Psi'\left(t_{n+\frac{1}{2}}\right)\Delta t + \mathcal{O}((\Delta t)^3)$$
$$\Psi'\left(t_{n+\frac{1}{2}}\right) = \frac{\Psi\left(t_{n+1}\right) - \Psi(t_n)}{\Delta t} + \mathcal{O}((\Delta t)^2)$$

The series for the right hand side of the PDE: $F(x,t)$, also centered around $t_{n+\frac{1}{2}}$ (fixed position):

$$F\left(t_{n+1}\right) = F\left(t_{n+\frac{1}{2}}\right) + F'\left(t_{n+\frac{1}{2}}\right)\frac{1}{2}\Delta t + \mathcal{O}((\Delta t)^2)$$
$$F(t_n) = F\left(t_{n+\frac{1}{2}}\right) - F'\left(t_{n+\frac{1}{2}}\right)\frac{1}{2}\Delta t + \mathcal{O}((\Delta t)^2)$$

Taking the sum of the series gives an approximation in second order of the timestep:

$$F\left(t_{n+\frac{1}{2}}\right) = \frac{1}{2}\left[F\left(t_{n+1}\right) + F\left(t\right)\right] + \mathcal{O}((\Delta t)^2).$$

With the left and right hand side computed, the original equation can be written with the finite difference in time as:

$$\frac{\partial\Psi(x_i, t_{n+\frac{1}{2}})}{\partial t} = F\left(x_i, t_{n+\frac{1}{2}}\right)$$
$$\Longrightarrow$$
$$\frac{\Psi_i^{n+1} - \Psi_i^n}{\Delta t} = \frac{1}{2}\left[F_i^{n+1} + F_i^n\right] + \mathcal{O}((\Delta t)^2),$$

where the shorthand notation $\Psi_i^n = \Psi(i\Delta x, n\Delta t)$ is used.

Finite differences for position

Taylor series for $\Psi(x_{i+1})$ and $\Psi(x_{i-1})$ near x_i.

$$\Psi(x_{i+1}) = \Psi(x_i) + \Psi'(x_i)\Delta x + \frac{1}{2}\Psi''(x_i)(\Delta x)^2 + \frac{1}{6}\Psi'''(x_i)(\Delta x)^3 + \dots$$
$$\Psi(x_{i-1}) = \Psi(x_i) - \Psi'(x_i)\Delta x + \frac{1}{2}\Psi''(x_i)(\Delta x)^2 - \frac{1}{6}\Psi'''(x_i)(\Delta x)^3 + \dots$$

From those we get:

$$\Psi(x_{i+1}) + \Psi(x_{i-1}) = 2\Psi(x_i) + \Psi''(x_i)(\Delta x)^2 + \mathcal{O}((\Delta x)^4),$$

which, after minimal manipulation, becomes a second order approximation for the second derivative with respect to position:

$$\Psi''(x_i) = \frac{\Psi(x_{i+1}) - 2\Psi(x_i) + \Psi(x_{i-1})}{(\Delta x)^2} + \mathcal{O}((\Delta x)^2).$$

Combining the finite differences

$$\begin{aligned} F(x_i,t) &= \frac{i\hbar}{2m}\frac{\partial^2}{\partial x^2}\Psi(x_i,t) - \frac{i}{\hbar}V(x_i)\Psi(x_i,t) \\ &\approx \frac{i\hbar}{2m}\frac{\Psi^n_{i+1} - 2\Psi^n_i + \Psi^n_{i-1}}{(\Delta x)^2} - \frac{i}{\hbar}V_i\Psi^n_i \\ &= \tilde{r}(\Psi^n_{i+1} - 2\Psi^n_i + \Psi^n_{i-1}) + \tilde{s}_i\Psi^n_i \\ &= F^n_i, \end{aligned}$$

with $\tilde{r} = \frac{i\hbar}{2m(\Delta x)^2}$ and $\tilde{s}_i = -\frac{iV_i}{\hbar}$. Taking the time discretised Scrödinger equation and plugging in the finite difference approximation for F:

$$\begin{aligned} \frac{\Psi^{n+1}_i - \Psi^n_i}{\Delta t} &\approx \frac{1}{2}\left[F^{n+1}_i + F^n_i\right] \\ &\approx \frac{1}{2}\left[\tilde{r}(\Psi^{n+1}_{i+1} - 2\Psi^{n+1}_i + \Psi^{n+1}_{i-1}) + \tilde{s}_i\Psi^{n+1}_i\right] \\ &\quad + \frac{1}{2}\left[\tilde{r}(\Psi^n_{i+1} - 2\Psi^n_i + \Psi^n_{i-1}) + \tilde{s}_i\Psi^n_i\right] \end{aligned}$$

It is convinient to move all the unknown $n+1$ terms on the opposite side as the known n terms, with $r = \tilde{r}\frac{\Delta t}{2}$ and $s_i = \tilde{s}_i\frac{\Delta t}{2}$:

$$-r\Psi^{n+1}_{i+1} + (1 + 2r + s_i)\Psi^{n+1}_i - r\Psi^{n+1}_{i-1} \approx r\Psi^n_{i+1} + (1 - 2r - s_i)\Psi^n_i + r\Psi^n_{i-1} = \psi_i.$$

This system of equations can be represented in matrix form:

$$\begin{bmatrix} a_0 & -r & 0 & \cdots & 0 \\ -r & a_1 & -r & \cdots & 0 \\ 0 & -r & a_2 & & \vdots \\ \vdots & \vdots & & \ddots & -r \\ 0 & 0 & \cdots & -r & a_I \end{bmatrix} \begin{bmatrix} \Psi_0^{n+1} \\ \Psi_1^{n+1} \\ \Psi_2^{n+1} \\ \vdots \\ \Psi_I^{n+1} \end{bmatrix} \approx \begin{bmatrix} \psi_0 \\ \psi_1 \\ \psi_2 \\ \vdots \\ \psi_I \end{bmatrix}, \tag{1}$$

where $a_i = 1 + 2r + s_i$. The matrix is tridiagonal and can be solved efficiently with the Thomas algorithm [2] with complexity $\mathcal{O}(I)$ as compared to the naive method of inverting the matrix that has complexity $\mathcal{O}(I^3)$, where I is the number of spatial gridpoints. By solving the system of equations, the values of the wavefunction at each of the gridpoints is obtained for the next timestep.

The Thomas algorithm consists of two parts that are executed in succession. In the first part the following quantities are calculated:

$$\lambda_i = \begin{cases} \frac{-r}{a_i}, i = 0 \\ \frac{-r}{a_i - r\lambda_{i-1}}, i \geq 1 \end{cases}$$

$$\psi_i' = \begin{cases} \frac{\psi_i}{a_i}, \ i = 0 \\ \frac{\psi_i + r\psi_{i-1}'}{a_i - r\lambda_{i-1}}, \ i \geq 1 \end{cases}$$

Using these quantities, the wavefunction at the next timestep is computed as:

$$\Psi_i^{n+1} = \begin{cases} \psi_I', \ i = I \\ \psi_i' - \lambda_i \Psi_{i+1}^{n+1}, \ i < I \end{cases}$$

References

[1] J. Crank and P. Nicolson, in *Mathematical proceedings of the Cambridge philosophical society*, Cambridge University Press (PUBLISHER , 1947), No. 1, pp. 50–67.

[2] W. Lee, MS6021, Scientific Computation, University of Limerick **1054**, (2011).